Contextualizing object detection and classification

Chen, Q., Song, Z., Dong, J., Huang, Z., Hua, Y., & Yan, S. (2015). Contextualizing object detection and classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(1), 13-27. https://doi.org/10.1109/TPAMI.2014.2343217

Published in:
IEEE Transactions on Pattern Analysis and Machine Intelligence

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

Contextualizing Object Detection and Classification

Qiang Chen, Zheng Song, Jian Dong, Zhongyang Huang, Yang Hua, and Shuicheng Yan

Abstract—We investigate how to iteratively and mutually boost object classification and detection performance by taking the outputs from one task as the context of the other one. While context models have been quite popular, previous works mainly concentrate on co-occurrence relationship within classes and few of them focus on contextualization from a top-down perspective, i.e. high-level task context. In this paper, our system adopts a new method for adaptive context modeling and iterative boosting. First, the contextualized support vector machine (Context-SVM) is proposed, where the context takes the role of dynamically adjusting the classification score based on the sample ambiguity, and thus the context-adaptive classifier is achieved. Then, an iterative training procedure is presented. In each step, Context-SVM, associated with the output context from one task (object classification or detection), is instantiated to boost the performance for the other task, whose augmented outputs are then further used to improve the former task by Context-SVM. The proposed solution is evaluated on the object classification and detection tasks of PASCAL Visual Object Classes Challenge (VOC) 2007, 2010 and SUN09 data sets, and achieves the state-of-the-art performance.

Index Terms—Object classification, object detection, context modeling

1 Introduction

Recognizing objects in an image requires combining many different signals from the raw image data. Two kinds of information are often used: the local appearance that describes the object itself and the global representation that captures the image specific information. These two types of information are often used in two tasks on visual recognition: object detection and classification. Object detection and classification are two key tasks for image understanding, and have attracted much attention in the past decade [18], [27], [41]. The object classification task aims to predict the existence of objects within images, whereas the object detection task targets to localize the objects. Several image databases tailored for these two tasks have been constructed, such as Caltech-101 [16]/256 [21], SUN data set [8] and PASCAL visual object classes (VOC) [15]. Many efforts [18], [27] have been devoted to these two tasks.

Beyond various image descriptors and modeling methods, the usage of context for visual recognition has become increasingly popular for enhancing the algorithmic performance. Many recent studies have demonstrated considerable improvements for object detection and classification by using external information, which is independently retrieved and complementary with traditional image descriptors. Specifically, the external context includes user-provided tags [4], [22], surrounding texts from Internet [2], [1], geo-tags and time stamps [14], etc. The context may also be the information lying within individual images. Intuitively, spatial location of the object and background scene from the global view can be used as intrinsic context of the image [24], [26].

We consider the context from the high-level task perspective. It has been demonstrated that the object detection and classification tasks can provide natural comprehensive context for each other without any external assistance, and thus can be mutually contextualized for performance boosting [23]. It is intuitively straightforward that for object classification task, the information from the local appearance promotes the performance significantly. For object detection task, the global context from object classification helps the detector better eliminate the false alarm. Although there are some works focusing on this direction, we notice that the underlying improvements brought by the context models for both two tasks have been underestimated. And the previous works take the context model in a multi-feature fusion fashion [4], [23] without dedicated design.

In this work, we develop a novel mutual contextualization scheme for object detection and classification based on the *Contextualized Support Vector Machine* (Context-SVM) method. First, we present a *contextualized learning* scheme via Context-SVM with the following characteristics:

- *Adaptive contextualization*. As many studies have shown [40], [38], context should be activated to be supportive mostly for those *ambiguous samples* and thus the context effectiveness should be conditional on the ambiguity of sample classification. The Context-SVM is superior over traditional learning schemes by complying with this principle in its formulation.
- *Multi-mode contextualization*. The ambiguity nature of the recognition problem at the boundary requires elegant design of the context model. We are interested in designing the localized context model along

- *Q. Chen is with IBM Research, Australia, and the National University of Singapore, Singapore. E-mail: qiangchen@au1.ibm.com.*
- *Z. Song, J. Dong, and S. Yan are with the National University of Singapore, Singapore. E-mail: {zheng.s, a0068947, eleyans@nus.edu.sg}.*
- *Z. Huang and Y. Hua are with the Panasonic Singapore Laboratories, Singapore. E-mail: {zhongyang.huang, yang.hua}@sg.panasonic.com.*

Manuscript received 28 Apr. 2013; revised 18 Nov. 2014; accepted 16 Feb. 2014. Date of publication 28 July 2014; date of current version 5 Dec. 2014. Recommended for acceptance by D. Ramanan.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2014.2343217

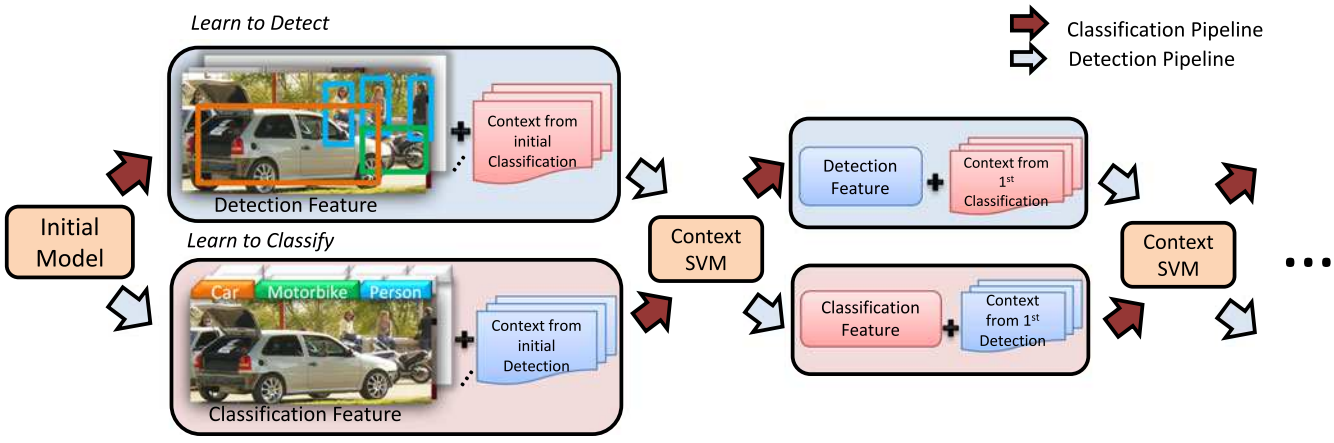


Fig. 1. Illustration of the iterative contextualizing procedure. The object detection and classification tasks utilize context from each other and mutually boost performance iteratively. For better viewing, please see original color PDF file.

the decision boundary which often shows various modalities. We propose to learn the multi-mode context model with mode selection function. Based on the general formulation, we further extend the context model to the ambiguity-guided mixture model (AMM). The mixture model naturally partitions the feature space at the decision boundary with regards to the ambiguity degree. Thus the proposed Context-SVM with multi-mode initialization can naturally embed the context model at the classification hyperplane.

- *Configurable model complexity*. The contextualization process should be efficient for both detection and classification tasks, and thus the solution should not involve many parameters. In this work, the Context-SVM with tractable control on the complexity of the context model is well formulated, so that the generalization capability is guaranteed.

Then we propose an iterative contextualization procedure based on the Context-SVM, such that the performance of object classification and detection can be iteratively and mutually boosted as illustrated in Fig. 1. Extensive experiments show that Context-SVM can efficiently learn the context models under various conditions and effectively utilize context information for performance boosting. We implement and evaluate the proposed scheme on object detection and classification tasks of the VOC 2007, VOC 2010 data sets [15] and SUN09 [8], and the results are superior over the state-of-the-art on most object categories.

An earlier version of this manuscript was presented as [37]. This version includes a clearer motivation section with a refined max margin model. Two ambiguity modeling methods are introduced with deeper analysis. Additional diagnostic experiments are conducted on both VOC and SUN09 data sets and new state-of-the-art results are presented. In the following, we first briefly review the related work for object recognition context modeling in Section 2. Then we introduce our ContextSVM model with two ambiguity modeling approaches in Section 3. Section 4 details our mutual and iterative contextulization for object detection and classification tasks. And we give extensive experiments on different data sets in Section 5.

2 Related Work

2.1 Context Modeling for Object Recognition

In recent years there has been a surge of interest in context modeling for numerous applications in computer vision. The basic motivation behind these diverse efforts is generally the same-attempting to enhance current image analysis technologies by incorporating information other than the image itself, e.g. semantic analysis result and metadata.

In the early work of Galleguillos and Belongie [20], the context refers to three main types of contextual information that can be exploited in computer vision: (1) the semantic context which refers to the likelihood of an object being found in some scenes but not in others, and from the point of view of modeling, can be expressed in terms of the corresponding object's probability of co-occurrence with other objects and the probability of occurrence in certain scenes; (2) the position (spatial) context which corresponds to the likelihood of finding an object in some positions and not others with respect to other objects in the scene; and (3) the size (scale) context which exploits the fact that objects have a limited set of size relations with other objects in the scene.

A natural way of representing the context of an object is in terms of its relationship with other objects, e.g. co-occurrence based context model [30]. An alternative terminology was proposed by Heitz and Koller [24] who introduced a "Things and Stuff" (TAS) context model. In their work, the terms "stuff" and "things" (originally introduced by Forsyth et al. [19]) are used to distinguish "material" that is defined by a homogeneous or repetitive pattern of fine-scale properties, but has no specific or distinctive spatial extent or shape (stuff) from "objects with specific size and shape" (things). Heitz and Koller claimed that "classifiers for both things or stuff can benefit from the proper use of contextual cues". Rabinovich and Belongie [33] proposed a classification of contextual models for computer vision (in general) and object recognition (in particular), consisting of models with contextual inference based on the statistical summary of the scene (which they referred as Scene Based Context models, SBC for short) and models representing the

context in terms of relationships among objects in the image (Object Based Context, OBC for short).

Also, some methods have been proposed to model the context in a comprehensive manner, e.g. [44], but they are quite specified and designed for one certain task, and thus cannot be generalized for our target in this work. We also notice that our work follows the research trend of stacking which which uses the output of the classifiers as the input for the next layer of classifier. e.g. Stacked Generalization [12]. More specifically, the auto-context model for image segmentation and labeling, e.g. AutoContext [43] and Texton [42]. The main difference of our work with those classifier combination methods is the usage of sample-specific ambiguity modelling.

Only recently, object hierarchy context has drawn much research attention [8], [47]. The object hierarchy is the further research of object co-occurrence context under the assumption that objects should be related with a semantic hierarchy. With the increased number of object categories, object relationship is naturally exhibited as a hierarchical structure. Context modeling with hundreds or thousands of object categories seeks to model this relationship with high level semantic structure or learned from data [11].

2.2 Mutual Contextualization for Object Classification and Detection

Although there are lots of works on context representation and modeling, few of them focus on contextualization between object detection and classification, namely, high level task context.

For object classification, the task cares more about whether the image contains a certain kind of object rather than where it is. The task is solvable due to the facts that (1) many data sets only concern the objects which occupy most of the images, e.g. Caltech 101 and 256 [16], (2) the same category objects often share similar scene level information, e.g. VOC and SUN09 data sets, and (3) the current prevalent object classification pipeline uses the sophistic feature encoding and learning method to extract image specific information which often reveals the object-specific contents, e.g. Fisher Vector Coding [32] and SVM classifier [3]. The methods used in classification are often built with a top-down manner that uses global information to infer the existence of a local object. For object detection, the task tries to localize the object within the image. Usually, the object detector models the object appearance [6] or object shape [10], [18] through the annotated object samples while discarding the context information defined by the object surrounding. The localized nature of the object detector restricts the model to effectively differentiate the false alarm which occurs at obviously different context. Harzallah et al. [23] introduced the pioneering work for object detection and classification contextualization through the post-processing of probability combination.

Moreover, traditionally, the context is considered as special features. Most of the existing strategies [14], [22], [23] utilize the context via feature concatenation, model fusion or confidence combination, and take the context as another independent component. However, context may have instable distribution, and its reliability and noise level are not controllable. Therefore it demands adaptive contextualization with proper constraints to avoid the inappropriate usage of context information. In this work, we follow this line to design the learning scheme for utilizing context information.

3 Contextualized SVM (Context-SVM)

In this work, the *context* is generally defined as certain extra supportive information for one task, which is retrieved independently from the *subject* task.[1] In this section, we first introduce the probabilistic motivation of the contextualized SVM and derive its linear formulation based on the probabilistic motivation. We then propose two ambiguity modeling methods for the Context-SVM which enables the multi-mode context modeling. Finally, we extend the linear Context-SVM to the kernel version for more general usage.

3.1 Probabilistic Motivation

Let $x_i^f \in \mathbb{R}^n$ denote the features of a sample for the subject task, $x_i^c \in \mathbb{R}^m$ denote the features of the corresponding context, and $y_i \in \mathbb{R}$ denote the ground-truth class label. Then the entire training data can be expressed as

$$\left\{x_i = \left\{x_i^f, x_i^c\right\}, y_i; i = 1, 2, \ldots, N\right\}. \tag{1}$$

Generally, the objective of a discriminative learning model can be defined to maximize the overall posterior probability on the training data:

$$\prod_{i=1}^{N} P(y = y_i \,|\, x_i).$$

There are two components within x_i, and usually the independent distribution of the subject features x_i^f and the context x_i^c is assumed and then $p(y \,|\, x_i)$ can be empirically modeled as:

$$p(y \,|\, x_i) = p\big(y \,|\, x_i^f\big) p\big(y \,|\, x_i^c\big). \tag{2}$$

The inference based on (2) is right for the traditional solution of confidence combination [14], [23] or multiple feature/model fusion [22].

The independence assumption, however, is often invalid for real data, and hence we propose to infer the label probability by (3) which explicitly models the conditional usage of context with respect to the given subject task, i.e., we empirically model

$$p(y \,|\, x_i) \approx p\big(y \,|\, x_i^f\big) \cdot p\big(y \,|\, x_i^c, x_i^f\big). \tag{3}$$

More specifically, we aim to infer the label probability via two components simultaneously. The first one tries to estimate the label probability based on the subject features, i.e. $p(y \,|\, x_i^f)$, and the second one is based on both the context and subject features. In practice, we relate the $p(y \,|\, x_i^c, x_i^f) \approx p(y \,|\, x_i^c, u(x_i^f))$ where $u(\cdot)$ is the ambiguous modeling function based on the subject feature so that the context

1. We refer the main/principal task concerned as the *subject* task.

modeling can be adopted to those ambiguous samples instead of the whole training set.

The ambiguity-based usage of the context information is critical for a contextualized learning. For object detection, the context of scene information from object classification is nearly the same for all detected windows within one image and may be unnecessary for many windows. For object classification, the context from object detection generally shows low reliability due to the possible false alarms and the selective usage of context can effectively avoid the disturbance caused by the false context to those already high-confident object patterns. Therefore we argue that only the ambiguous detection and classification results need the assistance from context.

3.2 Context-SVM: Formulation and Solution

3.2.1 General Formulation

For ease of formulation, we only consider the binary classification problem for object detection or classification task, i.e. $y_i \in \{+1, -1\}$ and the N_c-class problem can be decomposed into N_c binary classification problems through one-vs-all strategy. SVM [3] provides a general supervised learning framework by maximum margin optimization, and in this work, we extend SVM by introducing a novel parametrized model to describe the dependence between the context features and the subject features.

The general SVM learns a classifier over the subject feature space:

$$f(x_i, w_f) = {w_f}^T \cdot x_i^f + b, \tag{4}$$

and we can relate this scoring function $f(\cdot)$ with the log probability $\log p(y \mid x^f)$.

To utilize the extra supportive information from the context features x_i^c for the classification of x_i^f, the traditional combination of the subject and context features learns a classifier over the concatenated subject and context feature space:

$$f(x_i, w_f, w_c) = {w_f}^T \cdot x_i^f + {w_c}^T \cdot x_i^c + b, \tag{5}$$

which can be related to the value of $\log(p(y \mid x_i^f)p(y \mid x_i^c))$ using the independence assumption of the subject and context feature.

In our approach, we propose to utilize x_i^c with regards to the subject feature, namely an ambiguity-based classification. We first define the ambiguity modelling function $u(\cdot)$ which indicates the classification ambiguity in the subject feature space, and the ambiguity values (defined to be non-negative) along with the context feature x_i^c are then fed into the context models. To precisely model those ambiguous samples, we adopt a multi-mode context structure. There are totally R sub-context models are used. Thus $u(\cdot)$ is decomposed to R sub-models denoted as u_r, and each u_r is associated with a linear context model q_r as one component of the ambiguity-based context prediction. Consequently, the Context-SVM classification function can be denoted as

$$f(x_i, w_f, q_r) = {w_f}^T \cdot x_i^f + \sum_{r=1}^{R} u_r\big(x_i^f, w_f, \theta_r\big) q_r^T x_i^c, \tag{6}$$

where θ_r is the parameter associated with u_r. This Context-SVM model approximates our proposed joint subject and context model, i.e. the log probability $\log(p(y \mid x_i^f)\ p(y \mid x_i^f, x_i^c))$.

We denote the ambiguity values for sample i as $u_r(x_i^f, w_f, \theta_r) = u_{r,i}$, and then the Context-SVM scoring function can be expressed as

$$\begin{aligned} f(x_i, w_f, q_r) &= w_f^T x_i^f + \sum_{r=1}^{R} (u_{r,i} q_r)^T \cdot x_i^c + b \\ &\equiv w_i^T \cdot x_i + b, \end{aligned} \tag{7}$$

by defining $w_i = [w_0; u_{1,i}q_1; \ldots; u_{R,i}q_R]$ and $x_i = [x_i^f; x_i^c; \ldots; x_i^c]$. Here the defined w_i serves as a sample specific hyperplane which consists of the subject task model and R modes of context model parameters.

We then formulate the Context-SVM as a max-margin optimization problem with the margin described as the average of the rectified individual margins related to $\|w_i\|$'s, namely,

$$\begin{aligned} &\min_{w_0, \{q_r\}} \ \frac{1}{2N} \sum_{i=1}^{N} \|w_i\|_2^2 + C \sum_{i=1}^{N} \xi_i, \\ &s.t. \quad y_i\big(w_i^T x_i + b\big) - 1 + \xi_i \geq 0,\ \xi_i \geq 0,\ \forall i, \end{aligned} \tag{8}$$

where C is a tunable parameter for balancing two items and ξ_i are relaxation parameters.

Eq. (7) well shows the more insightful meaning of the Context-SVM formulation:

- The sample specific hyperplane w_i is the combination of the subject hyperplane w_f and R rectifications via $\{u_{r,i}, q_r\}$'s with the corresponding contributions determined by the context feature x_i^c. Intuitively, we can treat $u_{r,i}$ as a switch to determine whether the context should be activated while the value $q_r^T x_i^c$ determines how to rectify w_f.
- Motivated by probabilistic motivation (3), the $\{u_r(\cdot)\}$ and $\{q_r\}$ collaboratively describe one mode of the context model. $\{u_r(\cdot)\}$ serves to judge the discrimination ambiguity of x_i^f, and $\{q_r\}$ is utilized to integrate the context feature x_i^c for the classification of the samples with different ambiguities. The combination of R modes, each of which is composed of one $\{u_{r,i}, q_r\}$, enables the context model to approximate complex decision boundary.

3.2.2 Optimization for Context-SVM

It can be derived that Eq. (8) is equivalent to a standard SVM with regard to $v = [w_f; q_1; q_2; \ldots; q_R]$ with weighted regularization term.

Firstly, the regularization term of Eq. (8) can be reformulated as:

$$\begin{aligned} \frac{1}{2N} \sum_{i=1}^{N} \|w_i\|_2^2 &= \frac{1}{2} \|w_f\|_2^2 + \sum_{r=1}^{R} \frac{\lambda_r}{2} \|q_r\|_2^2 \\ &\equiv \frac{1}{2} v^T \Lambda v. \end{aligned} \tag{9}$$

Here $\lambda_r = \frac{1}{N}\sum_{i=1}^{N} u_{r,i}^2$ denotes the different regularization weight for the context model and $\Lambda = diag([I_n, \lambda_1 I_{n_c}, \ldots, \lambda_R I_{n_c}])$ is the entire diagonal weight matrix for v (n is dimension of the subject feature, n_c is the dimension of the context feature).

Then we format the soft margin term of Eq. (8) as:

$$y_i\big(v^T \hat{x}_i + b\big) - 1 + \xi_i, \tag{10}$$

and $\hat{x}_i = [x_i^f; u_{1,i}x_i^c; \ldots; u_{R,i}x_i^c]$ is a scaled feature vector.

Finally we combine Eqs. (8), (9), and (10) and obtain a weighted regularized SVM:

$$\begin{aligned} \min_{v} \quad & \frac{1}{2} v^T \Lambda v + C\sum_{i=1}^{N} \xi_i, \\ s.t. \quad & y_i\big[v^T \hat{x}_i + b\big] - 1 + \xi_i \geq 0, \quad \xi_i \geq 0, \quad \forall i. \end{aligned} \tag{11}$$

Note that in this optimization problem, there are only $(R \times n_c + n)$ parameters to optimize, and generally R and n_c is small. Therefore the overfitting issue can be well alleviated. Eq. (11) can be converted to a standard SVM problem and its solution can be derived with standard SVM solvers, e.g. LibSVM [5].

3.3 Ambiguity Modeling

In this subsection, we describe two methods to instantiate the $\{u_r(\cdot)\}$. As aforementioned, $\{u_r(\cdot)\}$ is the ambiguity selection function used to identify the ambiguity samples around the classification hyperplane so that finer classification is possible with the context feature. The flexible nature of Context-SVM allows us to instantiate the $\{u_r(\cdot)\}$ with multiple choices. Here we list two methods which we use in our experiments to instantiate the ambiguity selection function. The first one is the Linear Scaling Instantiation (LSI) which uses two linear scaling functions to select the ambiguity samples. The second one takes the estimation error of the original hyperplane as the ambiguity degree and then an Ambiguity-guided Mixture Model is learned. The corresponding $\{u_r(\cdot)\}$ severs as a context mode selection function at the decision boundary.

3.3.1 *Linear Scaling Instantiation*

As aforementioned, we design $\{u_r(\cdot)\}$ to highlight samples which are classified ambiguously with their subject features $\{x_i^f\}$. Practically, we instantiate $\{u_r(\cdot)\}$ as a set of scores with a learned hyperplane w_f in subject feature space by traditional SVM:

$$\begin{aligned} u_{r,i} &= u_r\big(x_i^f, w_f, \alpha_r, \beta_r\big) \\ &= \min\big(1, \max\big(0, \alpha_r w_f^T x_i^f + \beta_r\big)\big) \\ r &= 1, 2, \ldots, R. \end{aligned} \tag{12}$$

Intuitively, for $\alpha_r > 0$, if we set α_r and β_r properly such that all $\{u_{r,i}^T\}$ are within $[0, 1]$, those samples classified as negative by w_0 with high confidences shall be suppressed, namely their corresponding values of $\{u_r(\cdot)\}$ being small. At the same time, for $\alpha_r < 0$, if we set α_r and β_r properly such that all $\{u_{r,i}\}$ are within $[0, 1]$, those samples classified as positive by w_f with high confidences shall be suppressed, namely their corresponding values of $\{u_{r,i}\}$ being small. Therefore we can sample multiple combinations of α_r and β_r, and both strong negative and positive samples shall be suppressed by $\{u_r(\cdot)\}$ such that the samples with ambiguous decisions by w_0 are highlighted.

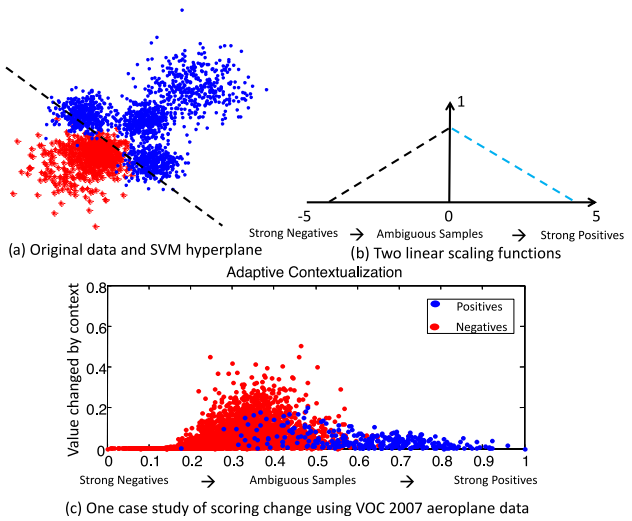


Fig. 2. Illustration of Linear Scaling Instantiation. a) The sample data with SVM hyperplane, red and blue dots representing positive and negative samples. b) The linear scaling functions. The black and blue dashed lines represent two different scaling functions. Each function scales one part of SVM scores with the range of $[0, 1]$. c) Illustration of the relationship between original sample confidence and confidence variation amount from context. The blue and red dots represent positive and negative samples respectively. The *x*-axis denotes the sample confidence in subject feature space and *y*-axis denotes the absolute amount of confidence changed by the contextualization procedure. The confidences are converted into probabilistic values within $[0, 1]$ indicating strongest negative and positive decisions respectively. For better viewing, please see original color PDF file.

More complicated $\{u_r(\cdot)\}$ with larger R may derive better ambiguity modeling but may also lead to overfitting. Our empirical study shows that it is a good trade-off by setting $R = 2$, i.e. using two auxiliary functions u_1 and u_2 where $\alpha_1 > 0$ and $\alpha_2 < 0$. Then the combination of $u_{1,i}$ and $u_{2,i}$ can provide a rough yet efficient judgement for the decision ambiguity of sample i and force the context model to concentrate on the samples with large ambiguities.

We illustrate one exemplar contextualization result by Context-SVM on object classification task of the "aeroplane" category in Fig. 2. This figure shows the adaptive contextualization with respect to the sample ambiguity: the output of the samples with higher ambiguities (i.e. samples lying in the middle of the figure) are changed (absolute difference value of the pre and after contextualization) largely by the contextualization procedure while the well-classified samples (i.e. samples lying on the two sides of the figure) are nearly unaffected.

3.3.2 *Ambiguity-Guided Mixture Model*

The flexibility of $\{u_r(\cdot)\}$ enables us to create the more complex context model near the classification boundaries. In the subject feature space, the ambiguous areas may be distributed in multiple localized areas and those areas naturally generate different modes. Thus an ambiguity-guided mixture model is necessarily learned to describe this ambiguity distribution. The local classifiers are then placed in areas

with high ambiguity. We first define the ambiguity degree a_i of a sample i as the hinge loss from the subject classification model:

$$a_i = max\big(0, 1 - y_i\big(w_f^T x_i^f + b\big)\big). \tag{13}$$

We propose the ambiguity-based mixture model for modeling the ambiguity distribution of the data. It is a mixture of R Gaussians, with each mixture component normally distributed as $N(\Sigma^r, \mu^r)$ with prior π^r, mean m^r and covariance matrix Σ^r. Assuming the parameter of the mixture model is ρ, the (combined) distribution function $p(x_i \,|\, \rho)$ at a particular sample x_i is the mixture probability. Obviously, the local classifiers should be placed near the decision boundary, where classification is the most difficult. Consequently, the mixture should have a high responsibility for areas with high uncertainties. In other words, $p(x_i \,|\, \rho)$ should be large when a_i is large, and vice verse. To achieve this goal, we maximize the following objective function:

$$F\big(a_i, x_i^f \,|\, \rho\big) = \sum_{i=1}^{N} a_i \log p\big(x_i^f \,|\, \rho\big). \tag{14}$$

And

$$\begin{aligned} u_{r,i} &= u_r\big(x_i^f, w_f, \rho\big) \\ &= p\big(r \,|\, x_i^f; \rho\big). \end{aligned} \tag{15}$$

We use expectation-maximization (EM) to optimize ρ:
E-Step

$$p\big(r \,|\, x_i^f\big) = \frac{\pi_r p\big(x_i^f \,|\, r; \rho\big)}{\sum_{r=1}^{R} \pi_r p\big(x_i^f \,|\, r; \rho\big)}. \tag{16}$$

M-Step

$$\mu^r = \frac{\sum_{i=1}^{N} a_i p\big(r \,|\, x_i^f\big) x_i^f}{\sum_{i=1}^{N} p\big(r \,|\, x_i^f\big) a_i}, \tag{17}$$

$$\Sigma^r = \left[\sum_{i=1}^{N} \frac{p\big(r \,|\, x_i^f\big) a_i}{\sum_{i=1}^{N} p\big(r \,|\, x_i^f\big) a_i} \big(x_i^f - \mu^r\big)\big(x_i^f - \mu^r\big)^T\right]^{-1}, \tag{18}$$

$$\pi^r = \frac{\sum_{i=1}^{N} p\big(r \,|\, x_i^f\big) a_i}{\sum_{r=1}^{R} \sum_{i=1}^{N} p(r \,|\, x_i^f) a_i}. \tag{19}$$

In practice, we notice that the dimensionality of x_i^f is often very high. The mixture model built upon this can be inaccurate. Thus we use principle components analysis (PCA) [25] to reduce the dimensionality, e.g. 512, while keeping the majority of data covariance.

We illustrate the concept of the ambiguity-guided mixture context model on a toy problem in Fig. 3. The red and blue dots on the left figure represent the positive and negative samples. The linear SVM hyperplane is illustrated by the black dashed line. It is obvious that linear SVM cannot get perfect separation on this data distribution. AMM models the ambiguity weighted data distribution. Each mixture

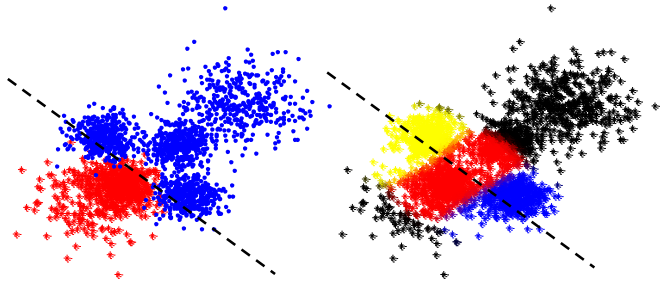


Fig. 3. Illustration of the ambiguity-guided mixture model on a toy problem. The left figure shows the original data. The red and blue dots represent the positive and negative samples. The linear SVM hyperplane is illustrated by the black dashed line. The right figure shows the AMM model with three mixtures (yellow, red and blue). It can be seen that the three mixtures are spreading over the hyperplane where the most ambiguous samples exist. The black dots represent the confidence samples which may not require the context model.

describes one local ambiguous area without considering the data distribution of the most confident samples. Thus the learned context model forming the localized classifier can better separate the data. The right figure shows the AMM model with three learned mixtures (yellow, red and blue). It can be seen that the three mixtures are spreading over the hyperplane where the most ambiguous samples exist. The black dots represent the confident samples which will not utilize the context model.

3.4 Kernel Extension

For many visual understanding problems, image descriptors are further encoded as similarity measurements or kernel matrices, and there is no explicit vector representation for each image. Therefore, it is necessary to generalize the Context-SVM formulation to the case with only kernel matrices available. It is worth noting that we only consider the subject feature in the kernel space. The context feature mentioned in this work is with low dimension and thus kernelization is not necessary. We consider the problem in the subject feature space $\mathcal{F}$ induced by a certain nonlinear mapping function $\phi : \mathbb{R}^n \rightarrow \mathcal{F}$. For a properly chosen ϕ, an inner product $\langle \cdot, \cdot \rangle$ can be defined on $\mathcal{F}$ which induces a Reproducing Kernel Hilbert Space (RKHS). More specifically, $\langle \phi(x_i^f), \phi(x_j^f) \rangle = \mathcal{K}(x_i^f, x_j^f)$ where $\mathcal{K}(\cdot, \cdot)$ is a positive semi-definite kernel function.

The context-adaptive scoring function for each sample can be defined as:

$$f(x_i, w_f, q_r) = w_f^T \phi\big(x_i^f\big) + \sum_{r=1}^{R} \big(u_{r,i} q_r^T\big) \cdot x_i^c + b, \tag{20}$$

which is similar to (7).

By Representer Theorem [36], w_f can be expressed as linear combinations of $\{\phi(x_j^f)\}, j = 1, 2, \ldots, N$ where N is the number of training data. Thus, there exist sets of coefficients such that $w_f = \sum_{j=1}^{N} \alpha_j \phi(x_j^f)$. Then, the scoring function can be expressed as:

$$f(x_i, w_f, q_r) = \sum_{j=1}^{N} \alpha_j \mathcal{K}\big(x_i^f, x_j^f\big) + \sum_{r=1}^{R} u_{r,i} \cdot \big(q_r^T x_i^c\big) + b. \tag{21}$$

Then the formulation can be compiled with respect to $\{\alpha_j\}$ and $\{q_r\}$as:

$$\begin{aligned} \min_{c} \quad & \frac{1}{2}c^T Bc + C\sum_{i=1}^{N}\xi_i, \\ s.t. \quad & y_i\left[c^T t_i + b\right] - 1 + \xi_i \geq 0, \quad \xi_i \geq 0, \quad \forall i, \end{aligned} \tag{22}$$

in which we define

$$\begin{aligned} B &= diag([I_N, \lambda_1 I_{n^c}, \ldots, \lambda_R I_{n^c}]), \\ c &= [\alpha_1; \ldots; \alpha_N; q_1; q_2; \ldots; q_R], \\ t_i &= \left[\mathcal{K}\left(x_i^f, x_1^f\right); \ldots; \mathcal{K}\left(x_i^f, x_N^f\right); u_{1,i}x_i^c; \ldots; u_{R,i}x_i^c\right]. \end{aligned} \tag{23}$$

The main differences between the kernel version and the linear version is the original subject feature vector is replaced by the kernel representation and eq. (22) is formulated with regard to α_j instead of w_f. Nevertheless, the same optimization approach can be used for solving the kernel extension of Context-SVM.

4 Application: Contextualizing Object Detection and Classification

In this section, we apply the Context-SVM to contextualize two prevalent tasks of image understanding, namely object detection and classification.

4.1 Initializations

The initial object detection and classification models $M_{det}(0)$ and $M_{cls}(0)$ for the first iteration are learned based on the state-of-the-art algorithms. For VOC data set, we follow the part-based model proposed by Felzenswalb et al. [18] for the initial detection model training. The histogram of gradient (HOG) [10] and local binary pattern (LBP) [29] features are used for object description and the number of part models for each object category is set as 6. For SUN09 data set, we use the newly proposed EMAS [6] object detection method due to its efficiency dealing with large number of categories.

For the object classification task, the traditional Bag-of-Words (BoW) model [17] is employed. We first extract the low-level features including SIFT and its color variants [39], LBP and HOG by dense sampling strategy in three scales. Each image is represented by BoW model with spatial pyramid matching [27]. The kernel function is based on χ^2 distance for each type of features, and then all kernels are combined as an average kernel for kernelized Context-SVM.

We define the subject feature as the raw feature for object classification and detection tasks, respectively. More specifically, the HOG and LBP feature are used as subject feature for object detection. For object classification, we utilize the kernel feature as the subject feature.

We derive the context feature from the classification and detection for the corresponding output. For the context feature from object classification task, we utilize the direct scores of the original classification model, e.g. BoW. For the context feature from object detection task, we select the top two detection scores for each detection models for one image and then concatenate them together to form the context feature. For example, on the VOC data set, we simultaneously train 20 classification and 20 detection models for the 20 object categories. Thus for each classification model on each image, it can get two scores from each detection models and hence a totally 40 dimension vector as the context feature. And for each detection task on each image sub-window, it can get 20 scores from totally 20 classification tasks as the context feature. Note that for all the detection samples from the same image, the context from the classification is the same.

4.2 Iterative Mutual Contextualization

The models output by the Context-SVM is still a linear SVM model and hence can be further contextualized. More specifically, the output model $v = [w_f; q_1; q_2; \ldots; q_R]$ from the Context-SVM can also be considered as a linear SVM model learnt on subject features $\hat{x_i} = [x_i^f; u_{1,i}x_i^c; \ldots; u_{R,i}x_i^c]$ $(i = 1, 2, \ldots, N)$. This property of the Context-SVM motivates us to propose an iterative contextualization procedure to further boost the object detection and classification performance.

The detailed algorithm for contextualizing object detection and classification by the iterative Context-SVM is shown in Algorithm 1. At the tth step, the subject features and context features of one task are obtained from the $(t-1)$th model on the training data. Note that we use cross validation method to obtain object classification scores on the training data since kernel model is easy to overfit on its training data. We use 10-fold of training data and evaluate each fold using the model trained on all other folds.

Algorithm 1. Contextualizing Classification and Detection

Input:

$M_{det}(0)$: Initial object detection model,
$M_{cls}(0)$: Initial object classification model,
I: Training images,
R: Ambiguity model complexity,
FUNCTION Subject(): Obtain subject detection and classification features,
FUNCTION Context(): Obtain the context detection and classification features,
FUNCTION Learn(): Obtain the context SVM model,
For $t = 1, 2, \ldots, T_{max}$

1) Obtain the subject feature

$$X_{det}^f(t) \leftarrow \text{Subject}(I, M_{det}(t-1))$$
$$X_{cls}^f(t) \leftarrow \text{Subject}(I, M_{cls}(t-1))$$

2) Obtain the context feature

$$X_{det}^c(t) \leftarrow \text{Context}(I, M_{cls}(t-1))$$
$$X_{cls}^c(t) \leftarrow \text{Context}(I, M_{det}(t-1))$$

3) Learning new Context-SVM model

$$M_{det}(t) \leftarrow \text{Learn}\left(X_{det}^f(t), X_{det}^c(t), M_{det}(t-1), R\right)$$
$$M_{cls}(t) \leftarrow \text{Learn}\left(X_{cls}^f(t), X_{cls}^c(t), M_{cls}(t-1), R\right)$$

EndFor
Output $M_{det}(T_{max}), M_{cls}(T_{max})$.

We instantiate $\{u_r(\cdot)\}$ based on the extracted subject features and the learnt model from the previous step. For Linear Scaling Instantiation, we use two linear functions to model the ambiguity, i.e. $R = 2$. One function is used to suppress the strong positive samples and the other is used to suppress the strong negative samples. For Ambiguity-guided Mixture Model, all the raw features are first reduced to 512 dimensions using PCA. A mixture model with $R = 20$ is constructed for each class.

5 Experiments

5.1 Data Sets and Metrics

The PASCAL Visual Object Classes Challenge data sets [15] are widely used as testbeds for evaluating algorithms for image understanding tasks, and provide a common evaluation platform for both object classification and detection. These data sets are extremely challenging since the objects vary significantly in size, view angle, illumination, appearance and pose. We use PASCAL VOC 2007 and 2010 data sets for experiments in this paper.

VOC 2007 and VOC 2010 data sets contain 20 object classes with 9,963 and 21,738 images respectively. The two data sets are divided into "train", "val" and "test" subsets, i.e. 25 percent for training, 25 percent for validation and 50 percent for testing. The annotations for the whole data set of VOC 2007 and "train", "val" set of VOC 2010 are provided while the annotations for "test" set of VOC 2010 are still confidential and can only be evaluated on the web server with limited trials. The employed evaluation metric is *average precision* (AP) and mean of AP (mAP) complying with the PASCAL challenge rules.

We also use the SUN 09 data set [8], which contains 4,367 training images and 4,317 testing images, for object classification and detection evaluation of 107 object categories. SUN 09 [8] has been annotated using LabelMe [35]. The author also annotated an additional set of 26,000 objects using Amazon Mechanical Turk to have enough training samples for the baseline detectors [18]. In the SUN09 data set, the average object size is 5 percent of the image size, and a typical image contains seven different object categories while the average PASCAL VOC bounding box occupies 20 percent of the image. These classes span from regions (e.g., road, sky, buildings) to well defined objects (e.g., car, sofa, refrigerator, sink, bowl, bed) and highly deformable objects (e.g., river, towel, curtain). The employed evaluation metric is *average precision* and mean of AP following [8].

In the following experiments, we first evaluate the mutual contextualization capability for ContextSVM with different ambiguity modelings (i.e. ContextSVM_LSI and ContextSVM_AMM) using VOC 2010 "train/val" data set (i.e. "train" set for training and "val" set for test) for both object classification and detection tasks for proof of concept and ease of parameter tuning. The iterative performance boosting is demonstrated in Section 5.3 on the VOC 2010 trainval/test data set. Then several traditional methods for contextualizing object detection and classification are compared with our iterative Context-SVM on the VOC 2010 trainval/test data set in Section 5.4. Finally, we evaluate the optimal configuration of our method on PASCAL VOC 2007, 2010 trainval/test data sets and SUN09 and compare with the state-of-the-art performance ever reported.

5.2 Mutual Contextualization

We first give the quantitative results for Context SVM on VOC 2010 train/val data set in Table 1 with one iteration setting. The improved results for object classification and detection tasks demonstrate the effectiveness of Context SVM.

For VOC 2010 classification task, we obtain the mAP of 0.681, a relative improvement of 12.56 percent over the classification baseline (0.605), with the context information from the detection raw results. The classification result shows the most improvement at those categories which often occupy small amount of the image space, e.g. bottle, tvmonitor, etc. We list some sample images improved by the contextualization as shown in Fig. 4. There are two rows showing the confidence change before and after the contextualization. The confidence has been normalized to $[0, 1]$. It is worth noting that the large changes are with those ambiguity samples whose original confidences are close to the 0.5. For example as shown in the first row, the third column of Fig. 4, the motorbike image has been classified with a confidence value of 0.41, and then the detection has a positive response within this image, so the final contextualized classification score for motorbike is very high. The contextualization for the classification task shows that the detection can be utilized to increase the recall rate of classification since the local model used by the detection task can find the objects occupying small part of the images.

For VOC 2010 detection task, we obtain the mAP of 0.327, a relative improvement of 15.55 percent over the detection baseline (0.283), with the context information from the classification results. The detection result shows the most improvement at those categories which often occupy large amount of the image space with large appearance variance, e.g. dogs, tables, etc. We list some sample images improved by the contextualization in Fig. 5. The role of the classification context model for detection tasks is mainly reflected by the fact that (1) the detection often fails for those samples with large appearance variance and the classification model is better to model the appearance changes, and (2) the local model used by detection tasks generally has no scene level context. In those two cases, the classification context model can help (1) to identify those objects with better appearance modeling and (2) to eliminate those false alarms by using the high level global context model. For example, as shown in the first row, the first two columns of Fig. 5, the classification context model helps to eliminate the false alarm detection of "tvmonitor" and further localize the true positive detection of "table".

Ambiguity modeling comparison. We give the quantitative results using different ambiguity modeling functions, i.e. Linear Scaling Instantiation and Ambiguity-guided Mixture Model. As shown in Table 1, both of these methods outperform the baseline methods with a large margin. Especially, AMM works better in terms of mAP. However, AMM does not outperform LSI at all 20 classes. Another observation is that for those classes with low AP accuracy, AMM performs similarly with LSI. It is reasonable since in that case the

TABLE 1
The Results of ContextSVM and Its Baseline for Object Detection and Classification Tasks on VOC 2010 Train/Val

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
cls_Baseline	86.9	59.1	61.7	68.3	29.9	82.9	63.3	70.1	55.2	36.4	50.6	53.4	53.7	64.1	84.5	36.1	60.1	49.1	79.2	66.2	60.5
cls_CtxSVM_LSI	**89.2**	72.8	64.7	72.5	**49.9**	87.4	77.3	74.3	62.5	40.1	49.6	58.5	66.3	73.1	91.4	41.8	64.8	52.8	84.7	**70.6**	67.2(+11.07%)
cls_CtxSVM_AMM	89.2	**73.0**	**66.1**	**73.4**	49.4	**87.9**	**78.1**	**75.4**	**63.3**	**40.8**	**50.9**	**59.3**	**69.0**	**74.8**	**91.7**	**46.4**	**65.5**	**53.6**	**85.2**	69.9	**68.1(+12.56%)**
det_Baseline	46.6	48.0	9.8	6.8	25.6	54.0	38.5	26.9	14.8	12.9	14.9	15.6	37.6	41.7	42.1	6.5	29.4	22.3	36.5	36.4	28.3
det_CtxSVM_LSI	50.2	49.3	16.8	11.6	27.0	55.4	39.8	**36.7**	16.4	17.7	19.8	23.1	41.0	44.4	**45.6**	**11.1**	**32.6**	**30.2**	39.3	**38.3**	32.3(+14.13%)
det_CtxSVM_AMM	**51.3**	**50.5**	**17.2**	**11.8**	**27.5**	**58.8**	**40.9**	35.0	**16.9**	**20.5**	**17.6**	**23.5**	**41.0**	**46.3**	45.4	10.1	29.7	28.3	**42.5**	38.1	**32.7(+15.55%)**

One iteration of ContextSVM is performed with two different ambiguity modeling methods, i.e. LSI and AMM. The relative improvement of mAP over the baseline without contextualization is also listed.

ambiguity modeling itself is not accurate. An analysis of AMM and LSI is as follows:

- The ambiguity modeling of LSI largely depends on the baseline prediction. It linearly scales the confidence obtained by the baseline and assigns higher values of $\{u_r(\cdot)\}$ to those ambiguous samples and lower values to those strong negative or positive samples. Then the learned context model q_r will act correspondingly.
- Unlike LSI, AMM models the data distribution as well as the baseline estimation. At the training stage of AMM, it incorporates the estimation error into the learning of the mixture model. The AMM learning concentrates on the data distribution of the ambiguous samples so that the learned mixtures better describe the complex decision boundary. The obtained $\{u_r(\cdot)\}$ corresponds to the posterior of sample i belonging to mixture r.
- The superiority of AMM over LSI probably comes from that (1) AMM considers the data distribution of ambiguous samples instead of only the baseline prediction in LSI, and (2) the number of mixtures in AMM is much larger than $R = 2$ in LSI. It is straightforward that a larger number of mixtures can better fit to the distribution of decision boundary, i.e. the ambiguous modeling. In all the experiments, we have fixed $R = 20$ in AMM as no obvious improvement can be observed when $R > 20$ from our offline experiments.

The role of contextualization. As shown in the results of VOC 2010 train/val data set, the Context SVM shows great improvement over the baseline for object detection and classification. Through the analysis and the experiments described above, it can be observed that it is necessary to use context for both object classification and detection tasks.

- For object classification, the prevalent methods [45], [31] use global features and discriminative modeling to achieve the goal. Although the current state-of-the-art recognition pipeline uses sophisticated feature encoding and learning methods to extract image specific information which often reveals the object-specific contents, e.g. Fisher Vector Coding [32] and SVM classifier [3]. The methods used in classification task are often built with a top-down manner which use global information to infer the existence of a local object. On the other hand, the context from the detection model contains rich local information. It greatly enhances the classifier to learn the image-specific information. As shown in Fig. 4, a lot of images containing small (or small-sized) objects are re-identified through contextualization.
- For object detection, usually the object detector models the object appearance [6] or object shape [10], [18] through the annotated object training samples while discarding the context information defined by the object surroundings. The localized nature of object detector restricts the model to effectively differentiate the false alarm which occurs in those obviously different contexts. The context information from classification model

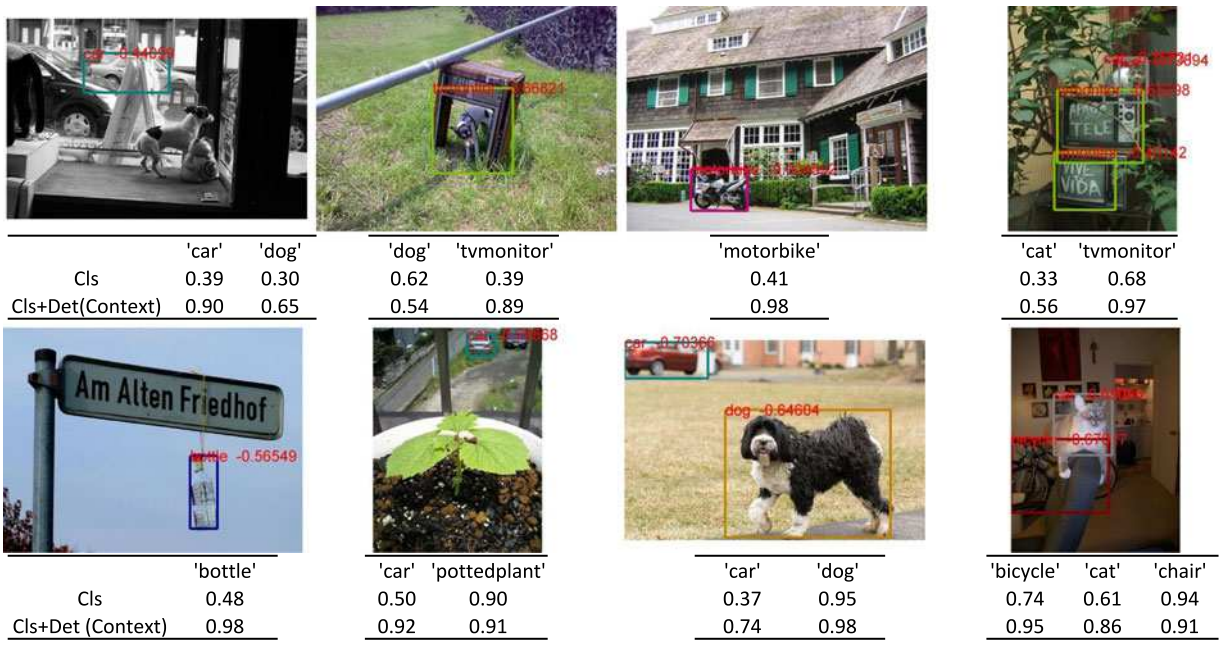


Fig. 4. Representative examples of the baseline (without contextualization) and Context-SVM for classification task. The classification accuracy is promoted via detection contextualization. The first row of the table below each image shows the classes the image belongs to. The second row is the confidence of the baseline while the third row is the refined result after contextualization. For better viewing, please see original color PDF file.

helps to define the context of the object. As shown in Fig. 5, it is helpful to eliminate the false alarm and promote the possible true positive.

- The ambiguity modeling enables that the learned context model concerns most on the ambiguous samples. The probabilistic motivation as introduced in Section 3.1 implies that it is desirable to learn the joint distribution of subject and the context feature instead of the independent learning as in [22], [23]. We propose to use the ambiguity modeling as a bridge between the subject and context task so that joint learning is possible. The learned context model operated on the ambiguous samples is better than the other context modeling method as demonstrated in Section 5.4.
- Another key advantage of conducting contextualization for both object classification and detection is that we can further build a more accurate context model with better classifier and detector through mutual contextualization. This step can be iterative until no further useful information can be learned as demonstrated in later Section 5.3.

5.3 Iterative Performance Boosting

To evaluate the effectiveness of our proposed iterative and mutual contextualization process, we conduct three experiments on VOC 2010 "train/val" data set. Firstly, we demonstrate the performance improvement measured by mean AP for all the 20 classes in Fig. 6. In this experiment, the mutual contextualization using LSI is conducted for three iterations, and obvious performance improvement is observed for the first and second iteration. As the improvement from the third iteration becomes trivial, we set the maximum iteration number, namely T_{max} to 3 for all the experiments in this work.

Fig. 5. Representative examples of the baseline (without contextualization) and Context-SVM for detection task. The detection accuracy is promoted via classification contextualization. The left side image is the result before Context SVM and the right side image is the result after contextualization. For better viewing, please see original color PDF file.

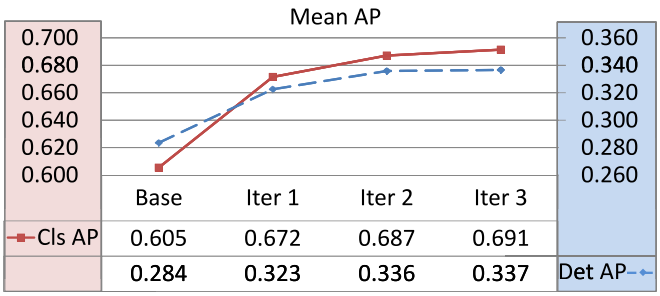


Fig. 6. Mean AP values of 20 classes on VOC 2010 train/val dataset along iterative contextualization.

In the second experiment, we show exactly how the mutual contextualization process benefits each class by Precision-Recall curves of several representative classes in Fig. 7, and also show the representative object detection and classification results in Fig. 8 for the third experiment. As can be observed from Fig. 7, great performance improvement can be achieved for the first two iterations and in the third iteration, certain amount of improvement can still be achieved for several classes such as "bus" and "dog". From Fig. 8, it can be observed that the Context-SVM shows good stability in refining the classes even without accurate context such as "pottedplant". The example detection results demonstrate that the improvement of object detection is mainly achieved by effective removal of the ambiguous negatives while the object classification benefits from detection context by calling back those missing objects, e.g. "person" and "chair" missed in the baseline results as shown in Fig. 8.

5.4 Contextualization Methods Comparison

In this subsection, we compare our proposed iterative and mutual contextualization method with other mutual classification and detection contextualization models. Three iterations of ContextSVM have been performed for each task.

We perform experiments on PASCAL VOC 2010 "trainval/test" data set and the results are shown in Table 2. We compare with the method proposed by Harzallah et al. [23] denoted as **Fuse**, which combines the confidences from several probabilistic models and is the most representative one among those confidence combination approaches [14], [18]. We also compare with the standard classifiers, e.g. SVM and Regression model (Reg) by concatenating the subject and context feature directly. We cannot obtain reasonable regression results for object detection task as it often involves large scale of training samples while regression-based model tend to "smooth" the prediction for the overall all training samples which makes it a bad detector. For object classification, Multiple Kernel Learning (MKL) [34] method used in [22] is also implemented for comparison, which is a general model fusion method and widely used to combine features in kernel form for object classification. An extra linear kernel is constructed for the context features from the object detection task, and then two kernels are combined with MKL. MKL performs badly for object detection task, and thus we do not report the result of MKL for object detection task here. The main reason is that the context is fixed for all candidate windows within an image and the inaccurate context may severely affect the results for quite many candidate windows. We also compare with SVM method using our iterative scheme denoted as SVM3, i.e. concatenate the output of each iteration and then retrain the model for three times. The main difference between our approach and SVM3 is our usage of ambiguous modeling. Such comparison directly reflects the usefulness of our approach.

The comparison results show that the proposed iterative and mutual contextualization method outperforms these contextualization methods for most object categories. The Reg baseline is much inferior to the SVM-based methods. We also observe that our ContextSVM is better than the SVM baseline. The inferior result of the SVM3 shows that our ambiguity modeling is essentially important for these complex classification problems. As illustrated in Fig. 3, our ContextSVM focuses on the ambiguous samples and thus

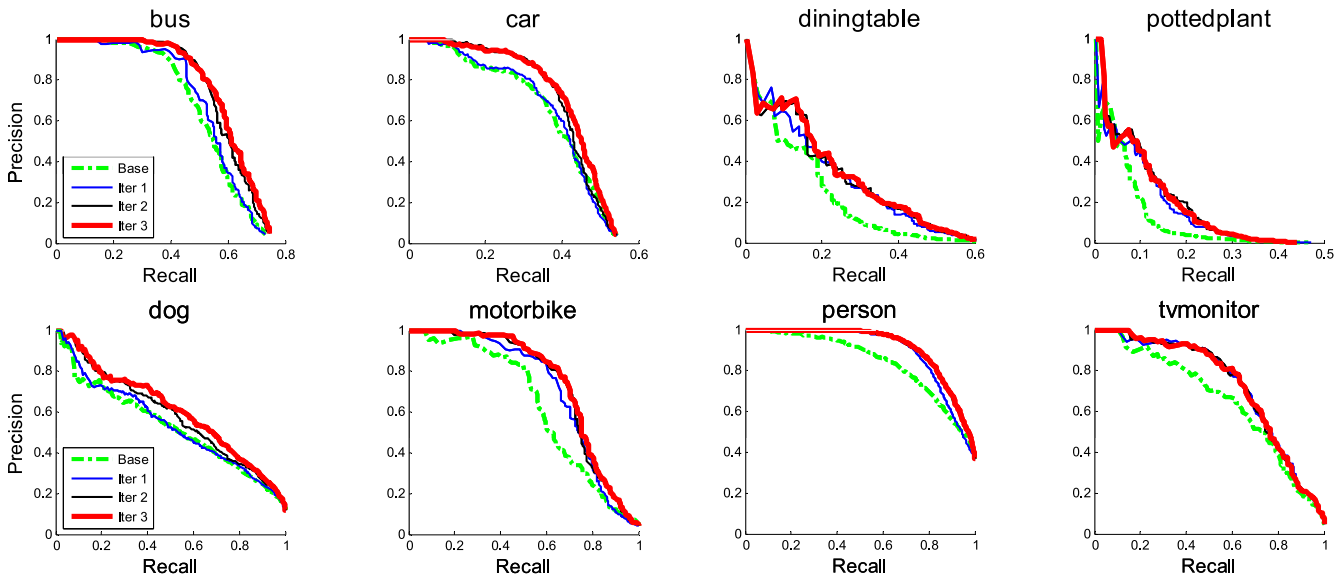


Fig. 7. Illustration of performance improvement with comparison Precision-recall curves of object detection (upper row) and classification (lower row). The performance of baseline (without contextualization) and those of Context-SVM at iteration 1-3 are plotted.

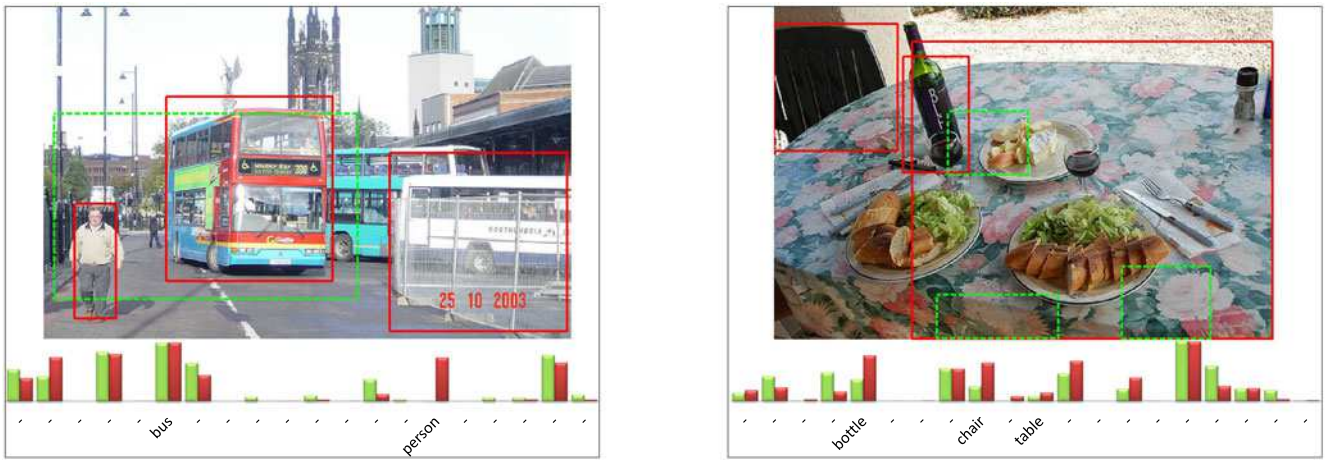


Fig. 8. Representative examples of the baseline (without contextualization) and Context-SVM at iteration 3. The detections are shown via the detected bounding boxes on images (with proper threshold): the green boxes with dashed lines denote the false alarms from baseline, which are further removed by contextualization and red boxes denote the true detections of both methods. The classification results are compared by the confidences for each object category before (green) and after (red) contextualization. For better viewing, please see original color PDF file.

the complementary effect from the context feature is well preserved. We also notice that AMM is consistently better than LSI for object classification task while achieving similar performance on the object detection task.

5.5 Comparison with State-of-the-Art Performance

We also compare the proposed contextualization method with the reported state-of-the-art object detection and classification approaches on VOC 2007, VOC 2010 and SUN09 data sets. The detailed performance comparison results are listed in Tables 3, 4, and 5.

We compare with the best known VOC 2007 performance from several recent papers in Table 3. For object detection, the methods compared include [MIT_2010] by Zhu et al. [46] using latent hierarchical structural learning, [UCI_2009] by Desai et al. [13] using context of object layout, [INRIA_2009] by Harzallah et al. [23] fusing classification scores, and [UoC_2010] by Felzenswalb et al. [18] using part-based model with context of object co-occurrence. For the detection challenge of 2007, our method outperforms 13 classes out of 20 classes and the MAP outperforms the second best [UoC_2010] by 3.6 percent.

The well-known methods compared for VOC 2007 object classification task are: [INRIA_Genetic] [28], the winner of VOC 2007, [NEC_2010] [45] performing nonlinear feature transformation on descriptors, [INRIA_2009] fusing detection scores, and [TagModal] [22] using extra tag information of VOC 2007 data set. Our method significantly outperforms the competing methods for 12 classes out of 20 classes. Note that our mAP (AMM 0.713) achieves a leading margin by 6.90 percent to the result of [TagModal](0.667). It well validates the effectiveness of the proposed strategy in utilizing detection context for object classification.

For VOC 2010 data set, we compare with the released results from the VOC 2010 challenge [15], which are all obtained through the combinations of multiple methods including mutual combination of detection and classification. Necessary postprocessing is also implemented in these methods. Therefore for a fair comparison, we refine the framework used by Chen et al. in their submission [NUSPSL] [7] with the following differences: 1) the combination of detection and classification is further refined by the proposed iterative Context-SVM and 2) we exclude the fusion of other learning schemes used in [7], e.g. the

TABLE 2
Contextualization Method Comparison on the PASCAL VOC 2010 (Trainval/Test) Data Set

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
Det Fuse	50.5	49.8	16.0	10.4	30.4	54.3	43.3	38.3	15.9	30.0	24.1	23.1	47.8	54.2	42.1	11.8	33.5	27.5	47.3	38.8	34.5
Det SVM	52.2	52.3	16.3	12.6	31.6	54.2	41.8	39.3	19.4	31.3	26.9	28.5	50.0	55.8	43.4	**13.2**	35.8	27.4	49.8	39.6	36.1
Det SVM3	52.7	52.4	16.8	12.6	**31.8**	**54.3**	41.3	39.1	**19.7**	31.4	26.6	28.4	50.2	55.6	43.8	13.1	**36.5**	27.1	50.0	39.2	36.1
CtxSVM_LSI	53.1	52.7	**18.1**	**13.5**	30.7	53.9	43.5	**40.3**	17.7	31.9	28.0	29.5	52.9	56.6	**44.2**	12.6	36.2	28.7	50.5	**40.7**	36.8
CtxSVM_AMM	**54.6**	**53.7**	16.2	12.5	31.2	54.0	**44.2**	40.0	16.7	**32.2**	**29.1**	**30.1**	**54.3**	**57.2**	43.9	12.5	35.4	**28.8**	**51.1**	**40.7**	**36.9**
	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
Cls MKL	91.4	76.6	66.7	72.3	53.1	83.7	77.1	75.3	62.9	59.8	57.1	63.6	76.5	81.8	91.2	44.1	64.1	48.4	84.0	75.5	70.3
Cls Fuse	90.7	74.0	67.2	73.9	53.8	81.7	74.1	73.6	60.9	59.8	60.5	62.3	75.1	80.2	90.4	45.8	61.7	56.0	85.9	76.0	70.2
Cls SVM	91.9	75.1	68.7	73.9	53.4	81.5	78.3	75.8	63.1	60.0	59.9	65.3	77.8	79.1	90.7	46.4	64.8	52.8	85.4	76.1	71.0
Cls SVM3	91.8	75.8	68.7	74.5	54.2	82.1	79.2	76.7	63.3	60.6	60.0	65.2	78.7	79.6	91.2	47.2	65.5	53.4	86.4	**77.0**	71.6
Cls Reg	86.4	72.3	65.3	71.8	52.2	82.8	70.2	74.7	55.3	53.9	56.2	60.4	72.4	76.6	86.7	42.4	60.9	50.0	82.2	73.9	67.3
CtxSVM_LSI	92.2	77.7	69.2	75.7	53.5	84.7	**80.9**	76.1	62.8	**65.5**	63.1	65.6	79.6	83.4	91.2	47.5	**71.9**	55.2	86.3	76.7	73.0
CtxSVM_AMM	**92.8**	**79.2**	**70.9**	**78.1**	**54.2**	**85.2**	78.9	**78.5**	**64.4**	64.5	**63.2**	**68.7**	**81.5**	**84.5**	**91.3**	**48.4**	65.0	**59.5**	**89.3**	76.0	**73.7**

"Det" and "Cls" respectively denote object detection and classification tasks. Three iterations of ContextSVM has been performed.

TABLE 3
Comparison with the State-of-the-Art Performance of Object Classification and Detection on PASCAL VOC 2007 (Trainval/Test)

	Detection on VOC 2007																				
	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
MIT_ZL [46]	29.4	55.8	9.4	14.3	28.6	44.0	51.3	21.3	20.0	19.3	25.2	12.5	50.4	38.4	36.6	15.1	19.7	25.1	36.8	39.3	29.6
UCI_ICCV09 [13]	28.8	56.2	3.2	14.2	29.4	38.7	48.7	12.4	16.0	17.7	24.0	11.7	45.0	39.4	35.5	15.2	16.1	20.1	34.2	35.4	27.1
INRIA_2009 [23]	35.1	45.6	10.9	12.0	23.2	42.1	50.9	19.0	18.0	**31.5**	17.2	17.6	49.6	43.1	21.0	**18.9**	**27.3**	24.7	29.9	39.7	28.9
UoC_04 [18]	31.2	**61.5**	11.9	17.4	27.0	49.1	**59.6**	23.1	23.0	26.3	24.9	12.9	60.1	**51.0**	**43.2**	13.4	18.8	36.2	49.1	43.0	34.1
CtxSVM_LSI	38.6	58.7	18.0	18.7	**31.8**	53.6	56.0	**30.6**	**23.5**	31.1	36.6	**20.9**	62.6	47.9	41.2	18.8	23.5	41.8	53.6	**45.3**	37.7
CtxSVM_AMM	**39.8**	59.0	**18.7**	**18.9**	30.0	**54.2**	57.2	30.4	**23.5**	30.9	**38.2**	20.7	**63.8**	48.8	41.5	18.7	23.8	**42.5**	**54.8**	44.9	**38.0**
	Classification on VOC 2007																				
	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
INRIA_Genetic [28]	77.5	63.6	56.1	71.9	33.1	60.6	78.0	58.8	53.5	42.6	54.9	45.8	77.5	64.0	85.9	36.3	44.7	50.6	79.2	53.2	59.4
SuperVec [45]	79.4	72.5	55.6	73.8	34.0	72.4	83.4	63.6	56.6	52.8	63.2	49.5	80.9	71.9	85.1	36.4	46.5	59.8	83.3	58.9	64.0
INRIA_2009 [23]	77.2	69.3	56.2	66.6	45.5	68.1	83.4	53.6	58.3	51.1	62.2	45.2	78.4	69.7	86.1	52.4	54.4	54.3	75.8	62.1	63.5
TagModal [22]	**87.9**	65.5	**76.3**	**75.6**	31.5	71.3	77.5	**79.2**	46.2	**62.7**	41.4	**74.6**	84.6	76.2	84.6	48.0	**67.7**	44.3	**86.1**	52.7	66.7
CtxSVM_LSI	82.5	79.6	64.8	73.4	**54.2**	75.0	**87.5**	65.6	62.9	56.4	**66.0**	53.5	**85.0**	76.8	91.1	53.9	61.0	67.5	83.6	70.6	70.5
CtxSVM_AMM	84.5	**81.5**	65.0	71.4	52.2	**76.2**	87.2	68.5	**63.8**	55.8	65.8	55.6	84.8	**77.0**	**91.1**	**55.2**	60.0	**69.7**	83.6	**77.0**	**71.3**

kernel regression fusing, to verify the effectiveness of the Context-SVM.

The comparison results are shown in Table 4, from which we may observe that the classification results from our proposed method outperform the others in 16 classes out of 20 classes, and 6.46 percent in terms of mean AP over the second best VOC 2010 submission [NLPR_Context]. Note that the submission [NLPR_Context] combines the best-performed detection results in this challenge for classification. Our proposed method also outperforms the winner submission [NUSPSL] in 17 classes out of 20 classes and achieves the highest mean AP even without the fusion with other learning methods. The object detection results from our proposed method based on Context-SVM also outperform seven classes out of 20 classes, and our method achieves the highest mean AP together with the winner submission [NLPR_Context], which outperforms six classes out of 20 classes in this competition.

We also conduct experiments on SUN09 data set [8]. The 107 classes mAP results on SUN09 data set for both object classification and detection tasks are listed in Table 5. The SUN 09 data set contains over 200 object categories but only 107 classes are used in [8] since some categories contain insufficient training samples. The baseline detectors of [8] for some objects have poor quality even with additional set of annotations. The current state-of-the-art performance is achieved in [8] which reported 8.55 for detection task and 26.08 for classification. In [8], the authors used a tree-based model to explore the hierarchical context between different objects. Compared with its baseline, the improvement of the TreeContext model is 3.82 percent (promoted from 7.06 mAP to 7.33) for object detection task and 11.15 percent (promoted from 19.93 mAP to 17.93) for object classification task. It further incorporates additional global features, i.e. gist feature, and context feature, i.e. location information to achieve the state-of-the-art performance with mAP 8.55 on the detection task. The other top performance is DPMContext which also used different scale and location information as the context feature.

We used the baseline of the EMAS object detector which shows great efficiency for object detection problem with a large number of object categories. EMAS performs better than the DPM [18] with 7.27 mAP for 107 classes while DPM reaches 7.06. The overall detection mAP over

TABLE 4
Comparison with the State-of-the-Art Performance of Object Classification and Detection on PASCAL VOC 2010 (Trainval/Test)

	Detection on VOC 2010																				
	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
NLPR [15]	53.3	**55.3**	**19.2**	**21.0**	30.0	54.4	**46.7**	41.2	**20.0**	31.5	20.7	30.3	48.6	55.3	**46.5**	10.2	34.4	26.5	50.3	40.3	36.8
MITUCLA [15]	54.2	48.5	15.7	19.2	29.2	**55.5**	43.5	41.7	16.9	28.5	26.7	30.9	48.3	55.0	41.7	9.7	35.8	30.8	47.2	40.8	36.0
NUS [15]	49.1	52.4	17.8	12.0	30.6	53.5	32.8	37.3	17.7	30.6	27.7	29.5	51.9	56.3	44.2	9.6	14.8	27.9	49.5	38.4	34.2
UVA [15]	**56.7**	39.8	16.8	12.2	13.8	44.9	36.9	**47.7**	12.1	26.9	26.5	**37.2**	42.1	51.9	25.7	12.1	**37.8**	**33.0**	41.5	**41.7**	32.9
CtxSVM_LSI	53.1	52.7	18.1	13.5	30.7	53.9	43.5	40.3	17.7	31.9	28.0	29.5	52.9	56.6	44.2	**12.6**	36.2	28.7	50.5	40.7	36.8
CtxSVM_AMM	54.6	53.7	16.2	12.5	**31.2**	54.0	44.2	40.0	16.7	**32.2**	**29.1**	30.1	**54.3**	**57.2**	43.9	12.5	35.4	28.8	**51.1**	40.7	**36.9**
	Classification on VOC 2010																				
	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
NLPR_Context [15]	90.3	77.0	65.3	75.0	53.7	85.9	80.4	74.6	62.9	66.2	54.1	66.8	76.1	81.7	89.9	41.6	66.3	57.0	85.0	74.3	71.2
NEC_Nonlin [15]	93.3	72.9	69.9	77.2	47.9	85.6	79.7	79.4	61.7	56.6	61.1	71.1	76.7	79.3	86.8	38.1	63.9	55.8	87.5	72.9	70.9
NUSPSL [15]	93.0	79.0	71.6	77.8	**54.3**	85.2	78.6	78.8	64.5	64.0	62.7	69.6	**82.0**	84.4	91.6	48.6	64.9	59.6	**89.4**	76.4	73.8
CtxSVM_LSI	93.1	78.9	73.2	77.1	**54.3**	85.3	80.7	78.9	64.5	68.4	64.1	70.3	81.3	83.9	91.5	48.9	72.6	58.2	87.8	76.6	74.5
CtxSVM_AMM	**93.8**	**80.5**	**74.7**	**78.3**	53.9	**86.5**	**82.4**	**80.3**	**64.9**	**72.8**	**65.7**	**73.3**	81.2	**85.3**	**91.8**	**50.2**	**72.9**	**61.6**	89.2	**77.2**	**75.8**

TABLE 5
mAP Results of 107 Classes on SUN09 Data Set for Both Object Classification and Detection Tasks

	Detection	Classification
Baseline_DPM	7.06	17.93
TreeContext [9]	7.33 (+3.82%)	19.93 (+11.15%)
TreeContext+loc+gist [9]	8.55 (+21.10%)	26.08 (+45.45%)
DPMContext [18]	8.34 (+18.13%)	23.79 (+32.68%)
Baseline_EMAS	7.27	22.23
CtxSVM_LSI	8.39 (+15.41%)	30.12 (+35.49%)
CtxSVM_AMM	8.56 (+17.74%)	31.43 (+41.39%)

The relative improvement of mAP over the baseline is also listed.

all object categories is 8.39 for the LSI instantiation and 8.56 for the AMM instantiation which leads to a 17.74 percent improvement. Our baseline of object classification has the result of mAP 22.23 which is slightly better than the result of [9]. Using the Context SVM, the performance with AMM instantiation can be boosted to 31.43 which is a 41.39 percent improvement over the original recognition score. Our implementation shows that we can achieve comparable state-of-the-art result with only the context from the high level task.

6 Conclusions

In this paper, we have proposed an iterative contextualization scheme to mutually boost performance of both object detection and classification tasks. We first propose the Contextualized SVM to seamlessly integrate external context features and subject features for general classification, and then Context-SVM is further utilized to iteratively and mutually boost performance of object detection and classification tasks. The proposed solution is extensively evaluated on both PASCAL VOC 2007, 2010 and SUN09 data sets and achieves the state-of-the-art performance for both tasks.

Acknowledgments

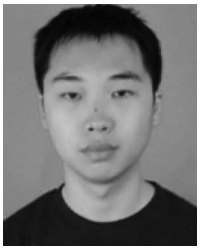
An earlier version of this work was appeared in CVPR'11.

References

[1] T. Berg, E. Berg, J. Edwards, and D. Forsyth, "Who is in the picture," in *Proc. Neural Inf. Process. Syst.*, 2006, pp. 264–271.
[2] T. Berg and D. Forsyth, "Animals on the web," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 1463–1470.
[3] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, pp. 121–167, 1998.
[4] P. Carbonetto, N. De Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 350–362.
[5] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 27, 2011.
[6] Q. Chen, Z. Song, R. Feris, A. Datta, L. Cao, Z. Huang, and S. Yan, "Efficient maximum appearance search for large scale object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3190–3197.
[7] Q. Chen, Z. Song, S. Liu, X. Chen, X. Yuan, T. Chua, S. Yan, Y. Hua, Z. Huang, and S. Shen, "Boosting classification with exclusive context," in *Proc. PASCAL VOC Challenge Workshop*, 2010.
[8] M. J. Choi, J. Lim, and Torralba,, "Exploiting hierarchical context on a large database of object categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 129–136.
[9] M. J. Choi, A. Torralba, and A. S. Willsky, "A tree-based context model for object recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 2, pp. 240–252, Feb. 2012.
[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 886–893.
[11] J. Deng, A. C. Berg, and L. Fei-Fei, "Hierarchical semantic indexing for large scale image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 785–792.
[12] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, pp. 241–259, 1992.
[13] C. Desai, D. Ramanan, and C. Fowlkes, "Discriminative models for multi-class object layout," in *Proc. Int. Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 229–236.
[14] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert, "An empirical study of context in object detection," in *Proc. IEEE Conf. Conf. Comput. Vis. Pattern Recog. Workshops*, 2009, pp. 1271–1278.
[15] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
[16] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Comput. Vis. Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.
[17] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 524–531.
[18] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[19] D. A. Forsyth, J. Malik, M. M. Fleck, H. Greenspan, T. Leung, S. Belongie, C. Carson, and C. Bregler, "Finding pictures of objects in large collections of images," EECS Dept., Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/CSD-96-905 Aug. 1996.
[20] C. Galleguillos and S. Belongie, "Context based object categorization: A critical survey," *Comput. Vis. Image Understanding*, vol. 114, no. 6, pp. 712–722, 2010.
[21] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category data sets," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2007-001, 2007.
[22] M. Guillaumin and J. Verbeek, "Multimodal semi-supervised learning for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010.
[23] H. Harzallah, F. Jurie, and C. Schmid, "Combining efficient object localization and image classification," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 237–244.
[24] G. Heitz and D. Koller, "Learning spatial context: Using stuff to find things," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 30–43.
[25] I. T. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer Verlag, Oct. 2002.
[26] S. Kumar and M. Hebert, "A hierarchical field framework for unified context-based classification," in *Proc. Int. Conf. Comput. Vis.*, 2005, pp. 1284–1291.
[27] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 2169–2178.
[28] M. Marszałek, C. Schmid, H. Harzallah, and J. van de Weijer, "Learning object representations for visual object class recognition," in *Proc. Vis. Recognit. Challenge Workshop Conjugation ICCV*, 2007, pp. 1–8.
[29] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognit.*, vol. 29, no. 1, pp. 51–59, 1996.
[30] A. Oliva and A. Torralba, "The role of context in object recognition," *Trends Cogn. Sci.*, vol. 11, pp. 520–527, 2007.
[31] F. Perronnin, Z. Akata, and Z. Harchaoui, "Towards good practice in large-scale learning for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 3482–3489.
[32] F. Perronnin, J. Sanchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 143–156.
[33] A. Rabinovich and S. Belongie, "Scenes vs. objects: A comparative study of two approaches to context based recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 92–99.

[34] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, 2008.
[35] B. Russell, A. Torralba, and K. Murphy, "LabelMe: A database and web-based tool for image annotation," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 157–173, 2008.
[36] B. Schölkopf, R. Herbrich, and A. Smola, "A generalized representer theorem," in *Proc. 14th Annu. Conf. Comput. Learn. Theory*, 2001, vol. 2111, pp. 416–426.
[37] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan, "Contextualizing object detection and classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1585–1592.
[38] A. Torralba, "Contextual priming for object detection," *Int. J. Comput. Vis.*, vol. 53, no. 2, pp. 169–191, 2003.
[39] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, Sep. 2010.
[40] L. Wolf and S. Bileschi, "A critical view of context," *Int. J. Comput. Vis.*, vol. 69, no. 2, pp. 251–261, 2006.
[41] J. Yang, K. Yu, and Y. Gong, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 1794–1801.
[42] J. Shotton and M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *Proc. IEEE Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[43] Z. Tu, "Auto-context and its application to high-level vision tasks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[44] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 17–24.
[45] X. Zhou, K. Yu, T. Zhang, and T. Huang, "Image classification using super-vector coding of local image descriptors," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 141–154.
[46] L. L. Zhu, Y. Chen, A. Yuille, and W. Freeman, "Latent hierarchical structural learning for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 1062–1069.
[47] A. Zweig and D. Weinshall, "Exploiting object hierarchy: Combining models from different category levels," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

Qiang Chen received the BE, MS, and PhD degrees from the Department of Automation, University of Science and Technology of China (USTC) in 2006, Department of Automation, Shanghai Jiaotong University in 2009, and ECE, National University of Singapore in 2013, respectively. He is currently a research scientist at IBM Research, Australia. He was a research fellow at the Electrical and Computer Engineering Department, National University of Singapore in 2013. His research interests include computer vision and pattern recognition. He received the Best Student Paper Awards PREMIA'12 and the winner prizes of the classification task in both PASCAL VOC'10 and PASCAL VOC'11, the honorable mention prize of the detection task in PASCAL VOC'10.



Zheng Song received the BSc degree from EECS Institute of Peking University in 2007, and the PhD degree from the Electrical and Computer Engineering Department, NUS, in 2013. He is currently a research fellow at School of Computing, National University of Singapore. His research interests include computer vision and intelligent systems.

Jian Dong received the BSc degree from the University of Science and Technology of China (USTC) in 2010. He is currently working toward the PhD degree with the Electrical and Computer Engineering Department, NUS. His research interests include computer vision and machine learning.

Zhongyang Huang received the bachelor's degree in biomedical engineering from Shanghai Jiaotong University, China, in 1993, and the master's degree in information engineering from Nanyang Technological University, Singapore, in 2001, respectively. From 1994 to 1999, he was a senior engineer for medical apparatus development in medical image processing area with China-America Joint Venture Kang Ming Biomedical Engineering Ltd. in China. Since 2001, he has been a senior staff R&D engineer of Panasonic Singapore Laboratories in Singapore. He received the winner prizes of the classification task in PASCAL VOC10 / VOC11. He has authored or coauthored one book chapter, 20 technical papers, and holds over 15 granted patents with numerous others pending in related fields.

Yang Hua received the bachelor's degree in electrical engineering and automation from China University of Mining and Technology and the master's degree in software engineering from Peking University in 2005 and 2008, respectively. Since July 2008, he has been with the Panasonic Singapore Laboratories in Singapore and he is currently a senior R&D engineer there. He was a joint winner of the classification task in PASCAL VOC 2010 and VOC 2011

Shuicheng Yan is currently an associate professor at the Department of Electrical and Computer Engineering, National University of Singapore, and the founding lead of the Learning and Vision Research Group (http://www.lv-nus.org). His research areas include computer vision, multimedia and machine learning, and he has authored or coauthored nearly 300 technical papers over a wide range of research topics. He is an associate editor of the *IEEE Transactions on Circuits and Systems for Video Technology (IEEE TCSVT)* and the *ACM Transactions on Intelligent Systems and Technology (ACM TIST)*, and has been serving as the guest editor of the special issues for TMM and CVIU. He received the Best Paper Awards from PCM'11, ACM MM '10, ICME'10 and ICIMCS'09, the winner prizes of the classification task in both PASCAL VOC'10 and PASCAL VOC'11, the honorable mention prize of the detection task in PASCAL VOC'10, 2010 TCSVT Best Associate Editor (BAE) Award, 2010 Young Faculty Research Award, 2011 Singapore Young Scientist Award, 2012 NUS Young Researcher Award, and the coauthor of the best student paper awards of PREMIA'09, PREMIA'11 and PREMIA'12.